

Dokumentacja

Funkcje przetwarzające dane rynkowe

Marcin Kolankowski

Dane rynkowe potrzebne do poprawnego działania funkcji

Dane rynkowe są umieszczone w pliku o nazwie `current_market_data.m`. Składają się one z kilku wartości skalarnych (dane identyfikujące, daty kwotowań, dane struktury kwotowań volatylity dla capów, dane struktury kwotowań volatylity dla swapcji) oraz tablic (macierzy) – DS, EC i ES.

dane identyfikujące – dane te zawierają dwie wartości: FC – nazwę centrum finansowego, z którego pochodzą dane oraz CURR – nazwę waluty, której dane dotyczą (zmienna CURR nie jest na razie wykorzystywana i przyjmuje ona stałą wartość $CURR = 1$).

daty kwotowań – daty z których pochodzą dostępne dane rynkowe. Daty te mówią z którego dnia pochodzą dane rynkowe użyte do dalszej estymacji parametrów. DS_date to data z której pochodzą czynniki dyskontowe, EC_date - data na którą są uzyskane kwotowania volatylity capów, ES_date - data uzyskania volatylity dla swapcji.

macierz czynników dyskontowych – macierz o oznaczeniu DS. Macierz ta zawiera dane dotyczące czynników dyskontowych. W pierwszej kolumnie są daty na które podane są czynniki dyskontowe (standardowa konwencja oznaczania daty). W drugiej kolumnie podane są czynniki dyskontowe na dane daty. W pliku znajduje się także data (DS_date), na którą zostały wyliczone dane w macierzy DS.

dane struktury kwotowań volatylity dla capów – na wielu rynkach kwotowania volatylity dla capów dla krótkich okresów dotyczą innej stopy LIBOR (3M) niż dla długich okresów (6M). Jeśli dane dla krótkich okresów są dostatecznie obszerne, można wyznaczyć volatylity dla krótszych okresów (dokładniejsza struktura czasowa volatylity). W tym celu definiowane są 2 wartości skalarne: "fine_cap_vol_ok", która jest indykatorem możliwości wyestymowania tej dokładniejszej struktury (fine_cap_vol_ok = 1) lub braku takiej możliwości (fine_cap_vol_ok = 0) oraz "fine_cap_vol_len", która podaje przez ile lat stosowana jest krótsza stopa.

macierz kwotowania volatylity dla capów – macierz o oznaczeniu EC, podaje kwotowania roczne volatylity dla capów. Kwotowania dla stopy forward 3M dla pierwszych dwóch lat, dla kolejnych okresów dane dotyczą stopy forward 6M. W kolumnach podano odpowiednio: czas mierzony w latach, aktualną stopę procentową, roczne volatylity dla capów ATM (szczegóły w książce Brigo - Mercurio s. 18). W kolejnych kolumnach podano volatylity dla innych wartości stopy strike.

dane struktury kwotowań volatylity dla swapcji – kwotowania volatylity dla swapcji dla krótkich okresów podawane są z krótczymi odstępami

czasu (co 3M a czasem dla zapadalności 1M). Jeśli dane dla krótkich okresów są dostatecznie obszerne, można wyznaczyć volatylity dla krótszych okresów (dokładniejsza struktura czasowa volatylity). W tym celu definiowane są 2 wartości skalarne: "fine_swap_vol_ok", która jest indykatorem możliwości wyestymowania tej dokładniejszej struktury (fine_swap_vol_ok = 1) lub braku takiej możliwości (fine_swap_vol_ok = 0) oraz "fine_swap_vol_len", która podaje dla ilu lat można zastosować tę dokładniejszą strukturę.

macierz kwotowania volatylity dla swapcji – macierz o oznaczeniu ES, podaje kwotowania roczne volatylity dla swapcji. Obowiązuje konwencja jak w książce Brigo-Mercurio str. 288 (w pierwszej kolumnie jest czas mierzony w latach do momentu wygaśnięcia opcji, w pierwszym wierszu są podane czasy życia swapu od momentu zapadnięcia opcji, tzw. tenory swapów, volatylity znajduje się w odpowiednim wierszu i kolumnie).

Czynniki dyskontowe i stopy forward

Będziemy oznaczać przez $DF(t, T)$ czynnik dyskontowy w chwili t liczony dla momentu T , tzn ilość kapitału jaki potrzebujemy w chwili t , aby przy strukturze terminowej stóp procentowych opisanej przez czynniki dyskontowe, w chwili $T > t$ uzyskać 1.

Czynnikiem dyskontowym forward $DF(t, T_0, T)$ będziemy nazywać czynnik dyskontowy obserwowalny w chwili t , który sprowadza do chwili T_0 przepływ pieniężny następujący w chwili T . Taki czynnik dyskontowy będziemy wyznaczać ze wzoru:

$$DF(t, T_0, T) = \frac{DF(t, T)}{DF(t, T_0)}.$$

Stopę forward w chwili t na okres (S, T) będziemy wyznaczać ze wzoru:

$$f(t, S, T) = \frac{1}{T - S} \left(\frac{DF(t, S)}{DF(t, T)} - 1 \right).$$

Funkcje wyznaczające krzywą dyskontową i stopy forward

DF_pom(T, conv)

Dane wejściowe: macierz DS wczytywana z pliku `market_data_xxxx.txt`;

Parametry funkcji: T (liczba), conv(string)

T – czas liczony w latach od daty DS_date;

conv – parametr wskazujący konwencję liczenia czasu pomiędzy dwoma datami: ACT/365, ACT/360, ACT/ACT, 30/360, 30E/360, opis szczegółowy przy funkcji year_frac;

Wynik działania funkcji: czynnik dyskontowy $DF(0, T)$;

Opis działania funkcji:

Funkcja korzysta z interpolacji oraz ekstrapolacji zadanej następującymi wzorami. Dla czasów T spełniających $T < T_0$ lub $T > T_n$ ekstrapolacja czynników dyskontowych wyraża się wzorem:

$$DF(0, T) = DF(0, T_0)^{\frac{T}{T_0}} \quad \text{dla } T < T_0,$$

$$DF(0, T) = DF(0, T_n)^{T/T_n} \quad \text{dla } T > T_n.$$

Dla czasów $T_0 \leq T \leq T_n$ leżących pomiędzy punktami $T_i \leq T \leq T_{i+1}$ używamy wzoru:

$$DF(0, T) = (DF(0, T_i))^{1-\tau} (DF(0, T_{i+1}))^\tau, \quad \tau = \frac{T - T_i}{T_{i+1} - T_i}.$$

DF(date0, dates, conv)

Dane wejściowe: macierz czynników dyskontowych jak przy funkcji DF_pom;

Parametry: date0 (data), dates (tablica), conv(string), curr (liczba);

date0 - data w konwencji dd-mm-yyyy oznaczająca datę na którą chcemy policzyć czynnik dyskontowy;

dates - w tablicy dates podajemy daty na które chcemy uzyskać czynniki dyskontowe;

conv - parametr wskazujący konwencję liczenia czasu pomiędzy dwoma datami: ACT/365, ACT/360, ACT/ACT, 30/360, 30E/360, opis szczegółowy przy funkcji year_frac;

Wynik działania funkcji: wektor czynników dyskontowych forward opisanych jak wyżej w części teoretycznej: (DF(DS_date, date0, dates{1}), ..., DF(DS_date, date0, dates{n}))

Opis działania funkcji:

Funkcja oblicza czynniki dyskontowe forward zgodnie ze wzorem zamieszczonym w części czynniki dyskontowe i stopy forward. Przykład wywołania funkcji:

```
x=DF("31-May-2010", {"17-Aug-2010", "20-Dec-2011"}, "ACT/365 ")
```

Funkcja zwróci nam wektor czynników dyskontowych dostępnych na dzień 31 maja 2010 r. Odwołanie x(1) zwraca nam czynnik dyskontowy z 17 sierpnia 2010 r. wyznaczony na dzień 31 maja 2010 r. Wybierając x(2) uzyskamy czynnik dyskontowy na dzień 31 maja 2010 r. z dnia 20 grudnia 2011 r. (wartość w dniu 31 maja 2010 r. 1 zł wypłaconej w dniu 20 grudnia 2011 r.). Funkcja korzysta z DF_pom oraz z funkcji kalendarzowej year_frac.

FR(date0, dates1, dates2, conv)

Dane wejściowe: macierz czynników dyskontowych jak przy funkcji DF_pom;

Parametry: date0 (data), dates1 (tablica), dates2 (tablica), conv(string);

parametry date0, conv jak w opisie funkcji DF;

dates1 - tablica złożona z dat w standardowej konwencji, oznaczających początki okresów na które chcemy mieć liczoną stopę forward;

dates2 - tablica złożona z dat w standardowej konwencji, oznaczających końce okresów na które chcemy mieć liczoną stopę forward;

Wynik działania funkcji: wektor złożony ze stóp prostych typu forward ($f(\text{date0}, \text{dates1}\{1\}, \text{dates2}\{1\}), \dots, f(\text{date0}, \text{dates1}\{n\}, \text{dates2}\{n\})$);

Opis działania funkcji:

Funkcja jako swoje parametry musi posiadać date0 - czyli datę z perspektywy której wyznaczane są stopy forward. Tablice dates1 i dates2 muszą być równoliczne, daty początku i końca okresu dla danej stopy powinny być na tych samych miejscach w dwóch tablicach. Zakładamy, że daty początku i końca okresu są w odpowiednim porządku. Stopy forward wyznaczamy ze wzoru zamieszczonego w części czynniki dyskontowe i stopy forward. Przykład wywołania funkcji:

```
x=FR("31-May-2010", {"31-May-2010", "13-Jan-2011"}, {"31-May-2011", "20-Aug-2011"}, "ACT/365 ")
```

Funkcja zwróci nam wektor stóp prostych forward na dzień 31 maja 2010 r. Odwołanie x(1) zwraca nam roczną stopę prostą. x(2) zwraca nam stopę forward obowiązującą pomiędzy datami 13 stycznia 2011 r. a 20 sierpnia 2011 r. wyliczoną w dniu 31 maja 2010 r.

Opis teoretyczny zastosowanych metod wyznaczenia volatylity

Teoria oraz metody tworzenia struktury volatylity pochodzą z książki Brigo-Mercurio *Interest Rate Models - Theory and Practice*. Jak już napisano przy opisie danych, volatylity dla capów o krótkich okresach odnosi się do capów na krótką stopę procentową (np. LIBOR 3M). Dodatkowo capy te są kwotowane z większą częstotliwością. Capy o dalekim horyzoncie zapadalności dotyczą dłuższych stóp procentowych (np. LIBOR 6M) i są kwotowane rzadziej. To pozwala wyestymowanie struktury czasowej volatylity na krótki okres z większą dokładnością. Stąd dla capów mamy dwie struktury volatylity: dokładniejszą na krótki okres i mniej dokładną dla długiego okresu. Należy także uważać na nieco inną konwencję gromadzenia danych w tabeli z volatylity, która w poszczególnych polach nie zawiera samych parametrów, a unormowaną całkę z $\sigma^2(t)$ po odpowiednim przedziale.

Metoda 1

Metoda 1 polega na przyjęciu stałej struktury volatylity dla danej stopy procentowej przez wszystkie okresy do momentu jej aktywowania. Opisana jest dokładniej jako metoda nr 3 w książce Brigo-Mercurio. Dzięki takiemu założeniu strukturę volatylity odczytujemy wprost z kwotowanej przez rynek volatylity

capów dla danego okresu. W tabeli zawierającej strukturę volatylity wyznaczoną przez metodę 1 (vol3m_1, vol6m_1) dane mają następującą postać:

okres	T_0	T_1	T_2	T_3	\dots
T_0	$\frac{1}{T_0} \nu_1^2$	0	0	0	\dots
T_1	$\frac{1}{T_1} \frac{T_0}{T_1} \nu_2^2$	$\frac{1}{T_1} \frac{T_1 - T_0}{T_1} \nu_2^2$	0	0	\dots
T_2	$\frac{1}{T_2} \frac{T_0}{T_2} \nu_3^2$	$\frac{1}{T_2} \frac{T_1 - T_0}{T_2} \nu_3^2$	$\frac{1}{T_2} \frac{T_2 - T_1}{T_2} \nu_3^2$	0	\dots
T_3	$\frac{1}{T_3} \frac{T_0}{T_3} \nu_4^2$	$\frac{1}{T_3} \frac{T_1 - T_0}{T_3} \nu_4^2$	$\frac{1}{T_3} \frac{T_2 - T_1}{T_3} \nu_4^2$	$\frac{1}{T_3} \frac{T_3 - T_2}{T_3} \nu_4^2$	\dots

gdzie zachodzi równość:

$$\nu_i = \nu_{cap}(T_{i-1}).$$

Tabelę należy czytać w następujący sposób. Stopa aktywna od momentu T_i (pierwsza kolumna tabeli) ma swoją volatylity opisaną w odpowiadającym wierszu. Unormowana przez czas volatylity dla poszczególnych podokresów znajduje się w kolejnych polach wiersza. Okres identyfikujemy przez czas jego końca, np. okres od chwili 0 do T_0 jest opisany w pierwszym wierszu macierzy jako T_0 . Stąd np. wartość volatylity dla danej stopy startującej od chwili T_i to pierwiastek z sumy elementów z danego wiersza macierzy.

Metoda 2

Metoda 2 ta polega na przyjęciu stałej volatylity zależnej tylko od czasu pozostałego do zapadalności. Metoda ta jest przedstawiona w książce Brigo-Mercurio jako metoda nr 2. Rozważmy przypadek kwotowań volatylity dla stóp forward 3M (np. dla okresu 2Y). Przyjmujemy, że $\tau = 1/4$ oraz $T_{-1} = 0$, $T_0 = 3M$, $T_1 = 6M$ itd. Niech η_j oznacza roczną volatylity dla okresu T_{j-1}, T_{j-2} . Z kwotowań rynkowych otrzymujemy, że volatylity dla początkowego okresu musi być stała, równa $\nu_{cap}(1Y)$ (jeśli brak danych poniżej 1Y, to $\eta_1 = \eta_2 = \eta_3 = \eta_4 = \nu_{cap}(1Y)$). Pozostałe η_i dla $i = 5, \dots, 8$ wyznaczamy z zależności:

$$\eta_i = 2 \cdot \sqrt{i \cdot \tau \cdot \nu_{cap}(T_{i-1})^2 - \tau \cdot \sum_{j=1}^{i-1} \eta_j^2}, \quad i = 5, \dots, 8.$$

Dla okresu większego niż 2Y volatylity capów podana jest dla stopy 6M. Przyjmijmy, że data kwotowania jest wyrażona na osi czasu przez 0. Teraz daty zapadalności oddalone są od siebie o 6M, stąd $T_{-1} = 0$, $T_0 = 6M$, $T_1 = 1Y$ itd, $\tau = 1/2$. W tym przypadku strukturę volatylity możemy wyznaczyć ze wzoru jak wyżej, modyfikując nieznacznie odpowiednie składniki:

$$\eta_i = \sqrt{2} \cdot \sqrt{i \cdot \tau \cdot \nu_{cap}(T_{i-1})^2 - \tau \cdot \sum_{j=1}^{i-1} \eta_j^2}, \quad i = 1, 2, \dots$$

Należy zauważyć, że stosujemy nieco inną konwencję gromadzenia danych w macierzy w porównaniu do książki. Oznacza to, że w tabeli przechowujemy nie

tyle same współczynniki η_i , ale w każdym wierszu mamy podaną unormowaną przez czas całkę z $\sigma_i^2(t)$ na odpowiednim odcinku czasu, równą po prostu η_i^2 , dokładniej:

$$\frac{1}{T_{j-1}} \int_0^{T_{j-1}} \sigma^2(t) dt = \frac{1}{T_{j-1}} \sum_{i=1}^j \eta_i^2,$$

co dla pojedynczego podokresu $0 \leq T_{k-1} < T_k \leq T_{j-1}$ oznacza, że:

$$\frac{1}{T_{j-1}} \int_{T_{k-1}}^{T_k} \sigma^2(t) dt = \frac{1}{T_{j-1}} \eta_{j-k}^2.$$

Ostatecznie tabela przedstawiająca volatylity (vol3m.2, vol6m.2) ma postać:

okres	T_0	T_1	T_2	T_3	\dots
T_0	$\frac{1}{T_0} \eta_1^2$	0	0	0	\dots
T_1	$\frac{1}{T_1} \eta_2^2$	$\frac{1}{T_1} \eta_1^2$	0	0	\dots
T_2	$\frac{1}{T_2} \eta_3^2$	$\frac{1}{T_2} \eta_2^2$	$\frac{1}{T_2} \eta_1^2$	0	\dots
T_3	$\frac{1}{T_3} \eta_4^2$	$\frac{1}{T_3} \eta_3^2$	$\frac{1}{T_3} \eta_2^2$	$\frac{1}{T_3} \eta_1^2$	\dots

Tabelę należy czytać w sposób następujący. Gdy chcemy dowiedzieć się jaka będzie volatylity dla stopy aktywnej od chwili T_i wybieramy cały wiersz z macierzy. Unormowana przez czas volatylity w poszczególnych podokresach do chwili T_i wyliczona jest w kolejnych polach danego wiersza, gdzie koniec okresu jest identyfikowany przez pierwszy wiersz macierzy. Np. wiersz:

okres	T_0	T_1	T_2	T_3	\dots
T_2	$\frac{1}{T_2} \eta_3^2$	$\frac{1}{T_2} \eta_2^2$	$\frac{1}{T_2} \eta_1^2$	0	\dots

opisuje volatylity dla stopy aktywnej od momentu T_2 , dla której w okresie $(0, T_0)$ unormowane przez czas volatylity wynosiło $\frac{1}{T_2} \eta_3^2$ itd.

Jeżeli chcemy podać kwotowania volatylity dla danej stopy startującej od chwili T_i wystarczy, tak jak w metodzie 1, wyciągnąć pierwiastek z sumy wszystkich elementów w danym wierszu macierzy.

Metoda 3

Ostatnia użyta metoda jest opisana w książce Brigo-Mercurio jako metoda nr 7. Polega ona na założeniu, że wielkość $\sigma_i(t)$, dla odpowiedniego okresu, ma parametryczną postać:

$$\sigma_i(t) = \Phi_i \left([a(T_{i-1} - t) + d] e^{-b(T_{i-1}-t)} + c \right).$$

Przy tym założeniu otrzymujemy, że:

$$\nu_i^2 = \Phi_i^2 \int_0^{T_{i-1}} \left([a(T_{i-1} - t) + d] e^{-b(T_{i-1}-t)} + c \right) dt = \Phi_i^2 I^2(a, b, c, d, T_{i-1}).$$

Aby ściśle dopasować się do danych rynkowych parametry Φ wyznaczamy ze wzoru:

$$\Phi_i^2 = \frac{(\nu_i^{MKT})^2}{I^2(a, b, c, d, T_{i-1})}.$$

Funkcje do wyznaczenia struktury volatylity

Volatility dla capów:

vol2vec()

Dane wejściowe: macierz kwotowań volatylity EC z pliku current_market_data.m oraz dane pomocnicze z tego pliku: EC_date, fine_cap_vol_ok, fine_cap_vol_len;

Wynik działania funkcji: dwie macierze z kwotowaniami volatylity dla capów 3M i 6M oraz zmienne EC_date, fine_cap_vol_ok znajdujące się w pliku volatility_vec.txt;

Opis działania funkcji:

Funkcja przekształca dane z macierzy EC do postaci dwóch macierzy: vol3m i vol6m. Każda z tych macierzy w pierwszej kolumnie ma czas mierzony w latach, w drugiej zaś odpowiadającą danemu czasowi volatylity. W macierzy vol3m czas podany jest w odstępach 1/4 (dla okresu, dla którego takie dane mają sens), volatylity podana jest dla stopy 3M. W macierzy vol6m czas podany jest w odstępach 1/2 i volatylity dla stopy 6M. Jest to techniczna funkcja przekształcająca dane do wygodniejszej postaci.

diff(param, volat)

Dane wejściowe: macierze volatylity z pliku volatility_vec.txt lub vol_swap;

Parametry: param (wektor), volat (macierz);

param - wektor zawierający parametry (a,b,c,d) służące do obliczenia błędu kwadratowego przybliżenia parametrycznego jak w opisie działania funkcji poniżej;

volat - macierz volatylity wytworzona przez funkcję vol2vec lub vol_swap2vec;

Wynik działania funkcji: funkcja zwraca wartość sumy kwadratów błędów pomiędzy wyznaczoną volatylity dla parametrów z wektora param, a volatylity znajdującą się w macierzy volat;

Opis działania funkcji:

Wprowadźmy następujące oznaczenie:

$$I^2(a, b, c, d, T_i) = \int_0^{T_i} \left([a(T_i - t) + d] e^{-b(T_i - t)} + c \right)^2 dt.$$

Funkcja zwraca nam błąd parametryzacji w stosunku do rynkowych danych. Oznaczając ν_i volatylity capleta dla okresu T_i , to wartości funkcji wyraża się wzorem:

$$\sum_i (I^2(a, b, c, d, T_i) - T_i \cdot \nu_i^2)^2$$

Funkcja techniczna, przydatna przy wyznaczeniu parametrów minimalizujących błąd kwadratowy.

find_param(vol)

Parametry: vol (macierz);

vol - macierz z kwotowaniem volatylity dla capów lub swapcji wyznaczoną przez funkcję vol2vec lub vol_swap2vec;

Wynik działania funkcji: parametry minimalizujące błąd kwadratowy dopasowania parametrycznego do danych z macierzy vol;

Opis działania funkcji:

Funkcja korzysta z metody Monte Carlo znalezienia parametrów minimalizujących błąd kwadratowy dopasowania parametrycznego do danych rynkowych. Dokonujemy $n = 100\,000$ losowań z rozkładu jednostajnego na $[-2, 2]$ każdego z parametrów niezależnie. Następnie każda tak wyznaczona czwórka $[a, b, c, d]$ jest podstawiana do funkcji znajdującej błąd kwadratowy dopasowania. Wybierane są parametry minimalizujące ten błąd.

find_int(a,b,c,d,x,z)

Parametry: liczby, które mają być parametrami do wyliczenia całki oznaczonej;

Wynik działania funkcji: liczba będąca całką oznaczoną z funkcji po danym przedziale;

Opis działania funkcji:

Funkcja wylicza wartość całki oznaczonej z funkcji po danym przedziale, a dokładniej:

$$find_int(a, b, c, d, x, z) = \int_0^x \left([a(z-t) + d] e^{-b(z-t)} + c \right)^2 dt$$

param_vol(a,b,c,d,z)

Parametry: liczby: a,b,c,d to parametry służące wyliczeniu struktury volatylity dla przedstawienia parametrycznego $I^2(a, b, c, d)$;

z - wektor czasowy dla którego chcemy mieć wyliczone wartości volatylity;

Wynik działania funkcji: wektor wyliczający volatylity wynikającą z parametrycznego założenia o wielkości $\sigma^2(t)$ jak w metodzie 3 dla czasów z wektora z;

Opis działania funkcji:

Funkcja wylicza parametryczną strukturę volatylity zgodnie ze wzorem:

$$\int_0^{T_i} \left([a(T_i - t) + d] e^{-b(T_i - t)} + c \right)^2 dt$$

dla poszczególnych okresów czasowych przekazywanych w wektorze z.

vol3m_estim()

Dane wejściowe: macierz vol3m wczytywana z pliku **volatility_vec.txt**;

Wynik działania funkcji: wynikiem działania funkcji jest plik o nazwie vol3m.txt w którym są trzy (lub dwie) macierze zawierające strukturę volatylity otrzymaną za pomocą dwóch (lub jeśli dane rynkowe pozwalają to trzech metod) oraz parametry dla parametrycznej struktury volatylity;

Opis działania funkcji:

Funkcja wylicza strukturę volatylity metodami opisanymi w części teoretycznej. Metoda 1 i 2 nie wymagają komentarza, funkcja wylicza tylko odpowiednie wielkości wprost ze wzoru. W przypadku metody 2 nie wszystkie dane rynkowe mogą zostać użyte (wynika to z przyjętych wzorów), dlatego na wstępie robiony jest test sprawdzający, czy dane rynkowe wyprodukują rzeczywiste parametry. Parametry wybieramy przy użyciu funkcji find_param. Po znalezieniu parametrów minimalizujących błąd kwadratowy, wyliczamy strukturę volatylity za pomocą funkcji param_vol. Następnie znajdujemy wektor Φ , który zgodnie z omówioną teorią ma zapewnić ściśle dopasowanie danych do kwotowania rynkowego w znanych punktach czasowych. Następnie tak otrzymane parametry z metody 1 i 2 (w postaci macierzy volatylity) oraz 3 (w postaci macierzy oraz otrzymanych parametrów) zapisywane są w pliku vol3m.txt. Macierz dla metody 1 ma nazwę vol3m_1, dla metody 2 - vol3m_2, dla metody 3 - vol3m_3. Pozostałe parametry mają odpowiednio nazwy: wektor Φ - phi, parametry - param. Dodatkowo zwracana jest stała met2_ok, której wartość jest równa 1, gdy z danych rynkowych można było utworzyć strukturę volatylity metodą nr 2. Gdy nie można było zastosować metody nr 2, zmienna met2_ok jest równa zero.

vol6m_estim()

Funkcja działa dokładnie tak samo jak opisana wcześniej funkcja vol3m. Jedyną różnicą jest taka, że otrzymujemy parametry dla stopy 6M oraz dotyczą one całego okresu dostępnych danych. Wynik działania funkcji znajduje się w pliku vol6m.txt. Poszczególne elementy mają odpowiednio nazwy: macierz dla metody 1 - vol6m_1, dla metody 2 - vol6m_2, dla metody 3 - vol6m_3. Pozostałe parametry mają odpowiednio nazwy: wektor Φ - phi6m, parametry - param6m. Dodatkowo zwracana jest stała met2_ok, która przyjmuje takie same wartości jak w opisie funkcji vol3m_estim().

vol_cap(date, dates, conv, freq, method)

Dane wejściowe: dane z plików zależnie od wyboru stopy: **vol3m.txt**, **vol6m.txt**;

Parametry funkcji: date (data), dates (tablica), conv (string), rate (liczba), method (liczba);

date - jest to data na którą chcemy uzyskać dane o volatylity - zazwyczaj data wyliczania ceny poszczególnych instrumentów;

conv - identyfikacja konwencji, jak przy wcześniejszych funkcjach;

dates - tablica złożona z dat zapadalności stóp forward (odpowiednich capletów

lub kuponów w swapie) dla których chcemy mieć volatylity;
 freq - zmienna identyfikująca częstotliwość podawanych wartości volatylity (dla capów to jest jednocześnie rodzaj stopy procentowej dla której liczymy volatylity) - 3 odpowiada częstotliwości 3M, 6 odpowiada częstotliwości 6M;
 method - wybór metody użytej do estymacji volatylity, nr metod zgodnie z opisem teoretycznym;

Wynik działania funkcji: funkcja zwraca wektor volatylity dla dat i stóp forward podanych w parametrach;

Opis działania funkcji:

Należy pamiętać, że volatylity dla stopy forward 3M jest kwotowana dla krótkiego okresu (zwykle do dwóch lat od daty uzyskania danych rynkowych) – mogą więc występować błędy w przypadku, gdy chcemy wykorzystać te dane dla stopy, która zapada poza okresem dostępnych danych. W takim przypadku należy korzystać ze stopy 6M. Parametr method identyfikuje metodę stosowaną do wyznaczenia volatylity. Metody mają takie same numery jak w opisie teoretycznym. W przypadku, gdy metoda 2 nie może być wybrana (brak danych rynkowych) automatycznie wybierana jest metoda 1.

Funkcja opiera się na wyliczeniu volatylity dla parametrów podanych przy wywołaniu funkcji i danych z macierzy vol3m i vol6m, zależnie od wybranej częstotliwości. Podstawą jest wyliczenie wektora x . Wektor ten służy do identyfikacji, jaka część volatylity z określonego okresu wchodzi w skład wyliczanej volatylity pomiędzy dwoma datami (wektor wag). Posiadając już taki wektor wyliczenie volatylity pomiędzy dwoma datami odbywa się przez pomnożenie odpowiednich współrzędnych - volatylity dla danego okresu oraz wagi z jaką ta volatylity wchodzi do volatylity całości.

Dla przykładu jak działa algorytm możemy rozważyć następującą sytuację: data kwotowania volatylity rynkowego - "08-May-2010", data wyliczenia volatylity dla stopy forward 3M - "21-May-2010", data zapadalności stopy "20-Nov-2010". W konwencji "ACT" mamy różnicę pomiędzy datami: 8 maja i 21 maja - 0.0356, 8 maja i 20 listopada - 0.537, 21 maja i 20 listopada - 0.501. Dodatkowo niech volatylity w poszczególnych okresach wynosi:

okres	0.25	0.5	0.75
	0.75	0.3	0.25
		0.2	

Stąd volatylity pomiędzy datami 21 maja i 20 listopada wynosi:

$$\nu = \frac{1}{0.501} \cdot \left(\frac{0.25 - 0.0356}{0.25 - 0.0} \cdot 0.3 + \frac{0.5 - 0.25}{0.5 - 0.25} \cdot 0.25 + \frac{0.501 - 0.5}{0.75 - 0.5} \cdot 0.2 \right).$$

Przykład wywołania funkcji:

x=vol.cap("14-May-2010 ", { "27-Jun-2010 ", "12-Dec-2011 " }, "ACT/365", 3,1)

Wynikiem działania funkcji jest volatylity roczna dla stopy 3M na dzień 14

maja 2010 r. $x(1)$ to volatylity dla stopy forward 3M zapadającej 12 czerwca 2010 r., liczonej za pomocą metody nr 1. Podobnie $x(2)$ to volatylity dla stopy forward 3M zapadającej 12 grudnia 2011 r.

Volatility dla swapcji:

Wyliczenie volatylity dla swapcji przebiega w taki sam sposób jak dla capów. Również korzystamy z trzech teoretycznych metod opisanych wcześniej. Wykorzystujemy także większą częstotliwość kwotowań dla krótszych okresów.

`vol_swap2vec()`

Dane wejściowe: macierz kwotowań volatylity dla swapcji ES z pliku `current_market_data.m` oraz dane pomocnicze z tego pliku: `ES_date`, `fine_swap_vol_ok`, `fine_swap_vol_len`;

Wynik działania funkcji: macierze z kwotowaniami volatylity dla swapcji dla częstości zapadalności 3M i 6M oraz odpowiednich tenorów zapisywane w pliku `vol_swap.txt`;

Opis działania funkcji:

Funkcja przekształca dane z macierzy ES do macierzy lepiej nadającej się do dalszej estymacji danych rynkowych. Wynikowe macierze zawierają volatylity kwotowane co 1/4 roku w tablicy `vol3m_swap` (dla krótszego okresu danych) oraz volatylity kwotowane co pół roku dla całego okresu dostępnych danych w tablicy `vol6m_swap`. Jest to techniczna funkcja przekształcająca dane do wygodniejszej postaci.

`vol3m_swap_estim()`

Dane wejściowe: macierz `vol3m_swap` wczytywana z pliku `vol_swap.txt`;

Wynik działania funkcji: wynikiem działania funkcji jest seria plików o nazwach `vol3m_swapx_1.txt`, ..., `vol3m_swapx_n.txt`, gdzie liczba "x_i" w nazwie pliku oznacza tenor swapu mierzony w latach. W każdym z takich plików znajdują się zmienne dokładnie takie same, jak w przypadku działania funkcji `vol3m_estim`;

Opis działania funkcji:

Funkcja działa analogicznie jak funkcja `vol3m_estim`, z tą różnicą, że dane wejściowe znajdują się w macierzy ES z pliku `current_market_data.m`. Wynikiem działania są opisane wyżej pliki. Każdy plik zawiera wyestymowane macierze volatylity różnymi metodami oraz parametry volatylity dla metody 3. W pliku o nazwie `vol3m_swapX.txt` są parametry volatylity dla swapcji o tenorze X. X musi być zgodne z tenorem danych rynkowych podanym w macierz ES (nie ma możliwości interpolacji dla tenorów niedostępnych w danych).

vol6m_swap_estim()

Dane wejściowe: macierz vol6m_swap wczytywana z pliku **vol_swap.txt**;

Wynik działania funkcji: wynikiem działania funkcji jest seria plików o nazwach vol6m_swapx_1.txt, ..., vol6m_swapx_n.txt, gdzie liczba "x_i" w nazwie pliku oznacza tenor swapu mierzony w latach. W każdym z takich plików znajdują się zmienne dokładnie takie same, jak w przypadku działania funkcji vol6m_estim;

Opis działania funkcji:

Funkcja działa analogicznie jak funkcja vol6m_estim, z tą różnicą, że dane wejściowe znajdują się w macierzy ES z pliku current_market_data.m. Wynikiem działania są opisane wyżej pliki. Każdy plik zawiera wyestymowane macierze volatylity różnymi metodami oraz parametry volatylity dla metody 3. W pliku o nazwie vol6m_swapX.txt są parametry volatylity dla swapcji o tenorze X. X musi być zgodne z tenorem danych rynkowych podanym w macierz ES (nie ma możliwości interpolacji dla tenorów niedostępnych w danych).

vol_swap(date, dates, conv, tenor, freq, method)

Dane wejściowe: dane z plików vol_swapX.txt, zależnie od wyboru tenoru;

Parametry funkcji: date (data), dates (tablica), conv (string), tenor (liczba), method (liczba);

date - jest to data na którą chcemy uzyskać dane o volatylity - zazwyczaj data wyliczania ceny poszczególnych instrumentów;

conv - identyfikacja konwencji, jak przy wcześniejszych funkcjach;

dates - tablica złożona z dat zapadalności swapcji dla których chcemy mieć volatylity;

tenor - zmienna identyfikująca tenor. Dostępne są tenory zawarte w danych ES;

freq - zmienna identyfikująca częstotliwość podawanych wartości volatylity – 3 odpowiada częstości 3M, 6 odpowiada częstości 6M;

method - wybór metody użytej do estymacji volatylity, nr metody zgodnie z opisem teoretycznym;

Wynik działania funkcji: funkcja zwraca wektor volatylity dla dat i swapcji o zadanych parametrach;

Opis działania funkcji:

Funkcja działa analogicznie do funkcji vol(), opisanej dokładnie powyżej.